Bartosz Łężniak 251574 (Lider)  
Maja Binkowska 251484  
Paweł Rajnert 251618

# Sterownik odtwarzacza MP3 na płytce LPC 1769

Zespół C10, czwartki godz. 16:00, tyg. Nieparzyste

Spis treści

1.

[Sterownik odtwarzacza MP3 na płytce LPC 1769 1](#_Toc194929399)

[1.1 Skład zespołu wraz z procentowym podziałem pracy 2](#_Toc194929400)

[1.2 Wykorzystane narzędzia 2](#_Toc194929401)

[1.3 Funkcjonalności 2](#_Toc194929402)

[1.4 Opis projektu 2](#_Toc194929403)

[2. Dokumentacja sterownika 3](#_Toc194929404)

[2.1 Instrukcja użytkownika 3](#_Toc194929405)

[2.2 Użytkowanie 3](#_Toc194929406)

[2.3 Algorytm działania 4](#_Toc194929407)

[3. Opis funkcjonalności 5](#_Toc194929408)

[3.1 Włączenie i wyłączenie odtwarzacza przez przycisk 5](#_Toc194929409)

[3.2 Wyświetlacz OLED 5](#_Toc194929410)

[3.3 SPI/F Wyświetlanie listy utworów (punkt kontrolny) 6](#_Toc194929411)

[3.4 Obsługa karty pamięci SD + SPI/F 6](#_Toc194929412)

[3.5 Przetwornik DAC 6](#_Toc194929413)

[3.6 Timer 6](#_Toc194929414)

[3.7 Przerwania 6](#_Toc194929415)

[3.8 Sterowanie wzmaczniaczem 6](#_Toc194929416)

[3.9 Obsługa pokrętła do wzmaczniacza 6](#_Toc194929417)

[3.10 Obrazowanie głośności na pasku LED 6](#_Toc194929418)

[3.11 Wykorzystanie I2C jako interfejsu do komunikacji z wyświetlaczem 6](#_Toc194929419)

## 1.1 Skład zespołu wraz z procentowym podziałem pracy

|  |  |
| --- | --- |
| Imię, Nazwisko i indeks | Udział procentowy |
| Bartosz Łężniak 251574 (Lider) | 34% |
| Maja Binkowska 251484 | 33% |
| Paweł Rajnert 251618 | 33% |

## 1.2 Wykorzystane narzędzia

|  |  |
| --- | --- |
| Nazwa sprzętu | Dodatkowe informacje |
| LPCXpresso LPC1769 | - |
| Karta SD 128 MB | System plików: FAT |

|  |  |  |  |
| --- | --- | --- | --- |
| Numer | Funkcjonalność | Status | Osoba odpowiedzialna |
| 1 | Włączenie i wyłączenie odtwarzacza przez przycisk | Działa | Bartosz Łężniak |
| 2 | Wyświetlacz OLED | Działa | Maja Binkowska |
| 3 | SPI/F Wyświetlanie listy utworów  (punkt kontrolny) | Działa | Paweł Rajnert |
| 4 | Obsługa karty pamięci SD + SPI/F | Działa | Bartosz Łężniak |
| 5 | Przetwornik DAC | Rozpoczęto | Paweł Rajnert |
| 6 | Timer | Nie rozpoczęto |  |
| 7 | Przerwania | Rozpoczęto | Bartosz Łężniak |
| 8 | Sterowanie wzmaczniaczem | Nie rozpoczęto |  |
| 9 | Obsługa pokrętła do wzmaczniacza | Nie rozpoczęto |  |
| 10 | Obrazowanie głośności na pasku LED | Nie rozpoczęto |  |
| 11 | Wykorzystanie I2C jako interfejsu do komunikacji z wyświetlaczem | Działa | Maja Binkowska |

## 1.3 Funkcjonalności

Kolejność powyższych funkcjonalności jest zgodna z harmonogramem zespołu C10 i nie odzwierciedla kolejności wykonywania poszczególnych etapów.

## 1.4 Opis projektu

Projekt został stworzony w języku C z wykorzystaniem płytki LPC 1769. Celem projektu jest podstawowa funkcjonalność odtwarzania muzyki w formacie MP3 zapisanej na karcie SD. Projekt obejmuje m.in. odczytywanie plików z zewnętrznego nośnika, wyświetlanie komunikatów na ekranie OLED, oraz odtwarzanie dekodowanego dźwięku przez głośnik.

# 2. Dokumentacja sterownika

## 2.1 Instrukcja użytkownika

Program służy do odtwarzania plików MP3 na dedykowanej płytce LPC 1769.  
Uruchomienie programu pozwala na korzystanie z poniższych funkcjonalności:

* **Odczyt plików MP3 z karty SD:** Przy uruchomieniu program automatycznie inicjalizuje interfejs SPI, sprawdzając poprawność połączenia karty SD, oraz pobiera i prezentuje podstawowe dane dotyczące jej parametrów. Jeżeli podczas odczytu karty SD wystąpi błąd, zostaje on wyświetlony na ekranie.
* **Prezentacja informacji na ekranie OLED:** Podczas inicjalizacji zostaje uruchomiony wyświetlacz OLED. Na wyświetlaczu pojawiają się komunikaty informujące o statusie systemu, szczegółach karty SD (takich jak liczba sektorów, rozmiar sektora, wielkość bloku oraz typ karty), a także lista wykrytych plików (maksymalnie dziewięć pozycji w jednym momencie).
* **Sterowanie przyciskiem:** Użytkownik ma możliwość włączania i wyłączania wyświetlacza OLED za pomocą dedykowanego przycisku. Naciśnięcie przycisku (wykrywane dzięki debouncingowi) powoduje włączenie lub wyłączenie ekranu. Zalecane jest wyłączenie ekranu w celu oszczędzania energii.

**Przed uruchomieniem programu**

1. Upewnij się, że karta SD jest sformatowana w systemie FAT z minimalnym rozmiarem alokacji.
2. W głównym katalogu karty SD umieść pliki MP3, które chcesz wczytać. Nie twórz podfolderów.
3. Sprawdź, czy wszystkie podzespoły (OLED, czujniki, przycisk) są poprawnie zamontowane i podłączone.

## 2.2 Użytkowanie

Program oferuje następujące możliwości:

* **Diagnostyka karty SD:**
  + W sytuacji gdy odczyt karty SD zakończy się niepowodzeniem, wyświetlacz OLED pozwala na odczytanie kodów błędu.
  + Zostają odczytane parametry karty, takie jak liczba sektorów, rozmiar sektora, wielkość bloku oraz określenie typu karty.
* **Wyświetlanie informacji:**
  + Na ekranie OLED pojawiają się informacje diagnostyczne dotyczące karty SD – są to m.in. dane techniczne oraz stan inicjalizacji, które potwierdzają gotowość karty SD.
  + Lista znalezionych plików MP3.
* **Interaktywne sterowanie wyświetlaczem:**
  + Program monitoruje stan przycisku odpowiedzialnego za włączanie/wyłączanie OLED’u.
  + Gdy zostanie wykryte wciśnięcie przycisku, mechanizm debouncingu przełącza stan wyświetlacza – przy włączeniu następuje ponowna inicjalizacja i odświeżenie danych, a przy wyłączeniu ekran jest czyszczony.

## 2.3 Algorytm działania

Poniżej przedstawione zostały główne etapy działania programu, krok po kroku:

1. **Inicjalizacja peryferii:**
   * **SPI:** Konfiguracja interfejsu SPI oraz odpowiednich pinów (SCK, MISO, MOSI) dla komunikacji z kartą SD. Dodatkowo, konfiguracja pinu GPIO służącego jako chip select (SSEL).
   * **I2C:** Ustawienie interfejsu I2C dla komunikacji z ekranem OLED, dzięki czemu urządzenie może przesyłać dane i konfigurować wyświetlacz.
   * **GPIO (przycisk):** Konfiguracja pinu przycisku, dzięki któremu użytkownik ma możliwość włączania i wyłączania ekranu OLED.
2. **Konfiguracja modułów systemowych:**
   * Wywołanie inicjalizujących funkcji modułów OLED, co sprawia że ekran jest w stanie gotowości.
   * Uruchomienie systemowego timera (SysTick) odpowiedzialnego za obsługę przerwań oraz realizację mechanizmu debouncingu.
3. **Obsługa karty SD:**
   * **Weryfikacja:** Program sprawdza status karty SD – jeśli wystąpią błędy (np. brak karty, błędne formatowanie), na ekranie wyświetlają się komunikaty o tym informujące.
   * **Odczyt parametrów:** Przy użyciu funkcji disk\_ioctl pobierane są kluczowe informacje dotyczące karty, takie jak liczba sektorów, rozmiar sektora, wielkość bloku oraz typ karty.
   * **Montaż systemu plików:** System plików FAT jest montowany, przez co można odczytać pliki z katalogu głównego i wykonywać operacje na plikach MP3 z karty.
4. **Wyświetlenie danych na OLED:**
   * Po inicjalizacji wyświetlacza, na ekranie OLED pojawiają się komunikaty przedstawiające wyniki testów karty SD oraz lista wykrytych plików.
   * Pliki są odczytywane z katalogu głównego karty, a program pomija wyświetlanie katalogów lub niewłaściwych plików.
5. **Główna pętla programu:**
   * Program monitoruje stan przycisku, stosując krótkie opóźnienie (debouncing) do zapobiegnięcia błędnych odczytów.
   * W momencie wykrycia opadającej krawędzi sygnału (naciśnięcia przycisku), następuje przełączenie flagi zarządzającej stanem wyświetlacza.
   * Jeśli OLED jest włączony, następuje jego ponowna inicjalizacja i wyświetlenie komunikatów diagnostycznych; w przeciwnym wypadku ekran zostaje wyczyszczony.

# 3. Opis funkcjonalności

## 3.1 Włączenie i wyłączenie odtwarzacza przez przycisk

## 3.2 Wyświetlacz OLED

Wyświetlacz OLED pełni funkcję wizualnej prezentacji informacji diagnostycznych i zawartości karty SD (np. listy plików MP3). Komunikacja odbywa się za pomocą interfejsu I2C2, co znacząco upraszcza układ, zmniejszając liczbę używanych pinów w porównaniu do rozwiązań opartych na SPI.

#### 3.2.1 Komunikacja

Wyświetlacz OLED komunikuje się z mikrokontrolerem przez interfejs I2C (opisany dalej w funkcjonalności 3.11), przez który konfiguracja sprzętowo jest znacznie prostsza, oraz pozwala na przesyłanie danych do wyświetlenia i komend sterujących.

#### 3.2.2 Sposób połączenia

Do skonfigurowania pinu P0.10 oraz P0.11 korzystamy z instrukcji *PinCfg.Pinnum()* na 10 lub 11 w zależności który pin konfigurujemy. Dalej używamy *PINSEL\_ConfigPin()* przypisując funkcję I2C.

* **Pin P0.10 (SDA: Serial Data):** dzięki temu pinowi OLED otrzymuje komendy i przesyłane informacje do wyświetlania obrazu. Od tego cały sterownik OLED interpretuje przesyłane bajty jako instrukcje dotyczące ustawienia kontrastu, trybu wyświetlania, czy wyczyszczenia ekranu.
* **Pin P0.11 (SCL: Serial Clock):** Ta linia odpowiada za zegar, który synchronizuje transmisję danych między mikrokontrolerem a wyświetlaczem. Dzięki temu sygnałowi OLED wie kiedy pobierać kolejne bajty danych.
* **Dodatkowa konfiguracja:** *I2C\_Init(LPC\_I2C2, 100000);* Inicjuje peryferium I2C2 z częstotliwością   
  100 kHz, przygotowując interfejs do komunikacji. *I2C\_Cmd(LPC\_I2C2, ENABLE)*; Aktywuje I2C2, umożliwiając rozpoczęcie transmisji danych.

#### 3.2.3 Inicjalizacja

Rozpoczyna się od uruchomienia funkcji oled\_init(), która wysyła do urządzenia sekwencję komend konfiguracyjnych. Wstawiono krótkie opóźnienie, przerwę w której urządzenie czeka na stabilizację sygnałów magistrali I2C. OLED później otrzymuje instrukcje dotyczące ustawienia trybu wyświetlania, konfiguracji kontrastu oraz sposobu adresowania pamięci wyświetlacza, co umożliwia przyjmowanie dalszych danych.

#### 3.2.4 Praca z ekranem

Po zakończeniu konfiguracji ekranu można przystąpić do wysyłania komunikatów na ekran. Korzystamy wtedy z funkcji *oled\_putString(),* która wysyła zawartość zmiennej typu string po magistrali I2C. Do wyczyszczenia ekranu stosujemy *oled\_clearScreen().*

* **oled\_putString(x, y, tekst, kolor\_txt, kolor\_tła):** odpowiada za wyświetlanie łańcucha znaków w określonym miejscu na ekranie. Parametr **x** oznacza pozycję poziomą, a **y** – pionową, gdzie tekst ma się zaczynać. Funkcja dalej korzysta z predefiniowanej czcionki, która przypisuje każdemu znakowi odpowiadający mu wzorzec pikseli. Piksele odpowiadające literom mają ustawiany kolor przez parametr **kolor\_txt**, a pozostałe są ustawione na kolor parametru **kolor\_tła**.
* **oled\_clearScreen(kolor):** funkcja iteruje po całym obszarze wyświetlacza piksel po pikselu, nadpisując jego zawartość jednakowym kolorem. W naszym kodzie na przykład wywołanie *oled\_clearScreen(OLED\_COLOR\_WHITE)* ustawia wszystkie piksele na biało, co usuwa wszelkie poprzednie rysunki lub tekst.

## 3.3 SPI/F Wyświetlanie listy utworów (punkt kontrolny)

## 3.4 Obsługa karty pamięci SD + SPI/F

## 3.5 Przetwornik DAC

## 3.6 Timer

## 3.7 Przerwania

## 3.8 Sterowanie wzmaczniaczem

## 3.9 Obsługa pokrętła do wzmaczniacza

## 3.10 Obrazowanie głośności na pasku LED

## 3.11 Wykorzystanie I2C jako interfejsu do komunikacji z wyświetlaczem

#### 3.11.1 Komunikacja

Wykorzystanie I2C jako interfejsu do komunikacji z wyświetlaczem OLED sprawia, że do prawidłowego działania potrzeba prostej konfiguracji w postaci jedynie dwóch pinów.

Piny **P0.10** i **P0.11** są odpowiedzialne za linie danych (SDA: Serial Data) i zegara (SCL: Serial Clock) interfejsu I2C2, a ich konfiguracja odbywa się przez funkcję *PINSEL\_ConfigPin()*, która ustawia je na alternatywną funkcję numer 2. Dzięki wywołaniu *I2C\_Init(LPC\_I2C2, 100000)*, interfejs zostaje zainicjalizowany z częstotliwością 100 kHz, co gwarantuje stabilną transmisję, a uruchomienie peryferium przez *I2C\_Cmd(LPC\_I2C2, ENABLE)* sprawia, że linie SDA i SCL stają się aktywne i gotowe do pracy. Szczegółowy opis pinów został także zamieszczony w sekcji *OLED 3.2.2 Sposób połączenia*.

#### 3.11.2 Inicjalizacja

Zaczyna się w funkcji *init\_i2c()*. Najpierw przy użyciu *PINSEL\_CFG\_Type* konfigurowane są piny **P0.10** oraz **P0.11** (za pomocą funkcji *PINSEL\_ConfigPin()*)*,* przypisując im funkcję interfejsu I2C (*FUNCNUM = 2*). Następnie, funkcja *I2C\_Init()* jest wywoływana z parametrem *LPC\_I2C2* i częstotliwością 100 kHz, co ustawia prędkość transmisji. Ostatecznie, funkcja *I2C\_Cmd()* włącza interfejs I2C, dzięki czemu linie **SDA** i **SCL** stają się aktywne i gotowe do obsługi komunikacji z OLED-em.

#### 3.11.3 Praca na fizycznym poziomie

Sterowanie magistralą I2C odbywa się przez bezpośrednią manipulację rejestrami I2C w mikrokontrolerze. Urządzenia są podłączone w sposób "open-drain" czyli mogą tylko ciągnąć linię do stanu niskiego, a stan wysoki osiąga się dzięki rezystorom podciągającym. W mikrokontrolerze funkcje takie jak *I2C\_Start()* i *I2C\_Stop()* generują sygnały rozpoczęcia i zakończenia transmisji, a dane wysyłane są przez zapis do specjalnego rejestru (I2DAT). Czas przez jaki linia zegara jest wysoka lub niska, ustalany jest przez rejestry I2SCLH i I2SCLL.

## 4. Bibliografia